



CarnegieMellon  
Software Engineering Institute

---

# Case Study: Building and Communicating a Business Case for a DoD Product Line

Sholom Cohen

*April 2001*

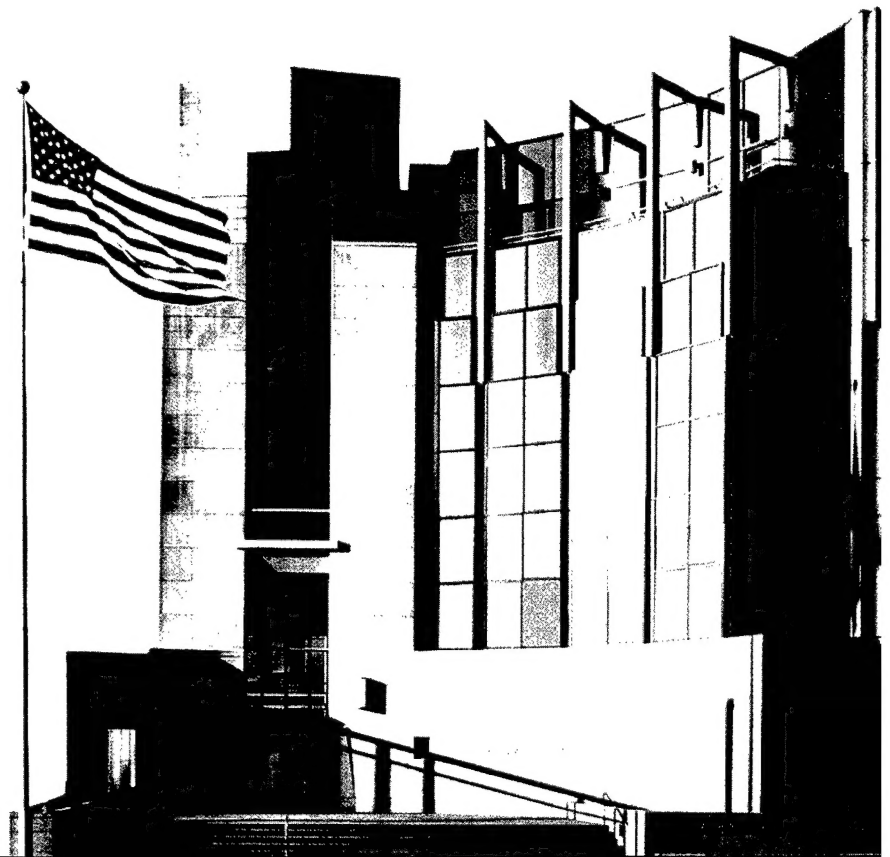
Product Line Practice Initiative

20011019 013

Unlimited distribution subject to the copyright.

Technical Note  
CMU/SEI-2001-TN-020

**DISTRIBUTION STATEMENT A**  
Approved for Public Release  
Distribution Unlimited



Carnegie Mellon University does not discriminate and Carnegie Mellon University is required not to discriminate in admission, employment, or administration of its programs or activities on the basis of race, color, national origin, sex or handicap in violation of Title VI of the Civil Rights Act of 1964, Title IX of the Educational Amendments of 1972 and Section 504 of the Rehabilitation Act of 1973 or other federal, state, or local laws or executive orders.

In addition, Carnegie Mellon University does not discriminate in admission, employment or administration of its programs on the basis of religion, creed, ancestry, belief, age, veteran status, sexual orientation or in violation of federal, state, or local laws or executive orders. However, in the judgment of the Carnegie Mellon Human Relations Commission, the Department of Defense policy of "Don't ask, don't tell, don't pursue" excludes openly gay, lesbian and bisexual students from receiving ROTC scholarships or serving in the military. Nevertheless, all ROTC classes at Carnegie Mellon University are available to all students.

Inquiries concerning application of these statements should be directed to the Provost, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, telephone (412) 268-6684 or the Vice President for Enrollment, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, telephone (412) 268-2056.

Obtain general information about Carnegie Mellon University by calling (412) 268-2000.

# **Case Study: Building and Communicating a Business Case for a DoD Product Line**

Sholom Cohen

*April 2001*

**Product Line Practice Initiative**

Unlimited distribution subject to the copyright.

**Technical Note**  
CMU/SEI-2001-TN-020

The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

Copyright 2001 by Carnegie Mellon University.

#### NO WARRANTY

THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this report is not intended in any way to infringe on the rights of the trademark holder.

Internal use. Permission to reproduce this document and to prepare derivative works from this document for internal use is granted, provided the copyright and "No Warranty" statements are included with all reproductions and derivative works.

External use. Requests for permission to reproduce this document or prepare derivative works of this document for external and commercial use should be addressed to the SEI Licensing Agent.

This work was created in the performance of Federal Government Contract Number F19628-00-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

For information about purchasing paper copies of SEI reports, please visit the publications portion of our Web site (<http://www.sei.cmu.edu/publications/pubweb.html>).

---

## **Contents**

<b>Acknowledgements</b>	<b>vii</b>
<b>Abstract</b>	<b>ix</b>
<b>1 Business Case Context</b>	<b>1</b>
1.1 Organizational Context	2
1.2 Refined SPO Goals and Alternatives	3
<b>2 Building the Business Case</b>	<b>4</b>
2.1 Establish Product Line Context	4
2.2 Estimate the Likely Costs and Analyze Potential Risks for All Alternatives	7
2.3 Estimate the Likely Benefits and Contrast With Current Business Practice	10
2.4 Develop a Proposal for Proceeding	12
2.5 Close the Deal: Make Final Adjustments and Proceed to Development	12
<b>3 Business Case Results</b>	<b>14</b>
3.1 Conclusions	14
3.2 Lessons Learned	14
<b>References</b>	<b>15</b>



---

## List of Figures

Figure 1: Coverage of Interfaces for Each System in Product Line	5
Figure 2: Coverage of Features for Each System in Product Line	5
Figure 3: Breakdown of Costs to Field the DoD Product Line	8
Figure 4: Product Line Versus Conventional Development Costs	9
Figure 5: Vision for Product Line	12





---

## List of Tables

Table 1:	Future Product Line Products	4
Table 2:	Candidate Alternative Courses of Action	7
Table 3:	Summary of Impact of Acquisition/Business Drivers on Candidate COAs	11



---

## **Acknowledgements**

I would like to acknowledge the contributions of Wolf Goethert and John Bergey to this technical note. Their work on the actual business case used in this note and their critiques have been most helpful.



---

## **Abstract**

This case study describes a US Department of Defense (DoD) weapon system development effort and compares the current way of developing software systems to the product line approach. The report describes a product line business case acquisition context, alternative development paths, and a comparison of their merits. While the report "sanitizes" the actual organization and product line, it typifies the process of building a product line business case.



---

## 1 Business Case Context

Software product lines<sup>1</sup> represent a significant business opportunity. Rather than build each product in stand-alone fashion, organizations use common assets to build related systems or sets of systems. This practice can yield quantifiable improvements in productivity, time to market, product quality, and customer satisfaction [Clements 00].

Before an organization moves to a product line approach for acquiring or developing software, it must address a number of key issues:

1. What constitutes the scope of the product line?
2. What similar products are available?
3. How many systems in the product line are likely to be fielded? Over what time period?
4. What are the costs involved in developing or mining assets?
5. What are the costs involved in producing systems in the product line?
6. What are the benefits of taking a product line approach? What are the risks?
7. How will the organization measure progress and success of the product line approach?

The answers to these questions will help management decide whether to launch a product line. An effective business case must demonstrate that the investment is financially sound, that it is aligned with business strategies, that it can be implemented within acceptable time and cost parameters, and that it will provide the benefits intended.

This report describes a business case undertaken by a US Department of Defense (DoD) acquisition organization. Section 1 presents the organizational context. Section 2 provides the steps followed in building the business case. Section 3 describes lessons learned.

To maintain anonymity, this report refers to the DoD organization as the "SPO."

---

<sup>1</sup> A *software product line* is a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way [Clements 00].

## 1.1 Organizational Context

The SPO currently supports a number of related weapons systems that may be grouped together as a DoD product line. The systems in the DoD product line (System A, System B with its variants B1 and B2, and System C) all deliver roughly the same capability to users. They differ, however, in major ways:

- The systems' platforms evolved from previous weapons systems. Each legacy system has maintained its heritage computing environment while evolving to deliver roughly the same functionality.
- The number of features varies among the systems. System A provides the most features. System B1's features are a proper subset of System A's. System B2 includes all of System B1 along with some features not in A. There are also other variants in planning: System C will be derived from System B2.
- The number and types of external interfaces vary among the systems. The systems work with a wide range of peripherals, including special purpose processors, communications equipment, and other weapons systems. For the most part, all systems in the DoD product line share interfaces, though System A and variants of System B have different implementations of those interfaces.
- The current architecture for System A is unique. The variants of System B share a common architecture and the planned System C will likely evolve from the architecture of System B. The common architecture may diverge as future systems evolve. Software code is not shared between System A and the variants of System B, even software that supports interfaces to the same equipment.
- Separate development organizations are responsible for each system.
- Users are familiar with the characteristics of existing Systems A, B1, and B2. The SPO wishes to continue these as separate systems, maintaining existing user interfaces for the next generation to avoid retraining.

Before generating new systems or new versions of existing systems, the SPO evaluated several different strategies:

- Use current methods of developing or acquiring systems to add capabilities.
- Redesign current systems, retaining separate architectures and maintenance requirements.
- Migrate all systems to a common architecture, retaining unique developers. Under this approach, it may be possible to share some implemented changes.
- Build a product line around a common architecture and systematic reuse of components. Use product line assets to develop new versions of Systems A, B, and C, as well as all future systems in the DoD product line. Assign one or more developers to generate assets and field individual DoD product line systems.



## **1.2 Refined SPO Goals and Alternatives**

The SPO faced rising costs and anticipated the obsolescence of computing platforms for its legacy systems. Rather than maintain separate development programs, the SPO saw that, in general, a product line approach could help it

- Reduce costs.
- Increase reliability.
- Simplify and speed the integration of new interfaces.

However, the SPO also recognized that there were risks involved in changing its current way of working. To assess these risks, as well as the relative costs and benefits, the SPO developed a business case. It compared the SPO's current production method to alternative product line strategies.

The SPO established the business case around the five phases described by Humphrey [Humphrey 00]:

1. Establish a product line context. The business case team listed assumptions (market conditions, organizational goals, etc.), developed alternative approaches, and compared those approaches.
2. Estimate the likely costs and analyze potential risks for all alternatives.
3. Estimate the likely benefits contrasted with current business practice.
4. Develop a proposal for proceeding.
5. Close the deal. Make final adjustments and proceed to development.

Section 2 covers each of these five phases.

---

## 2 Building the Business Case

### 2.1 Establish Product Line Context

The product line approach must maintain the current high standards of system performance, while reducing the acquisition cycle. However, the conversion to a product line approach takes time, so its benefits may not be apparent, or available, immediately. Given this understanding and the SPO's desire to achieve goals quickly, the business case team examined the current acquisition approach and alternative product line strategies. At the same time, the business case team worked with other teams already looking at product line scope and goals.

A business case operates under a set of assumptions. To develop these assumptions, the team determined the scope of the product line, analyzed possible future products, and reviewed organizational goals.

The product line scope covers a set of services (or features) that address product line requirements, the systems themselves, and interfaces to other systems that must be supported. In this instance, the SPO determined that it will field three systems (Systems A, B, and C) in the product line over the next five years. In addition, the SPO will acquire two variants of System B, namely B1 and B2.

The following table describes the product line scope by summarizing the systems and their respective capabilities. Even though the product line scope only reflects a high-level estimate, it is sufficient for building the business case.

*Table 1: Future Product Line Products*

System Version	System Capabilities and Feature Coverage
System A	Full set of interfaces; 80% of all features within product line scope
System B, variant 1 (B1)	80% of interfaces; 60% of all features within product line scope
System B, variant 2 (B2)	Same interfaces as System B, variant 1; all features of System B, variant 1, plus features within product line scope not in System A
System C	40% of interfaces; scaled back version of System B, variant 2

Another way to scope the product line is to examine interface and feature coverage in each system. Table 2 shows that all systems contain a basic set of interfaces, while Systems A and both B variants include support for additional interfaces. Similarly, all systems share a set of basic features, as shown in Figure 2. Some systems also contain extensions. Again, the estimates of 100% coverage are only for establishing upper limits in the business case.

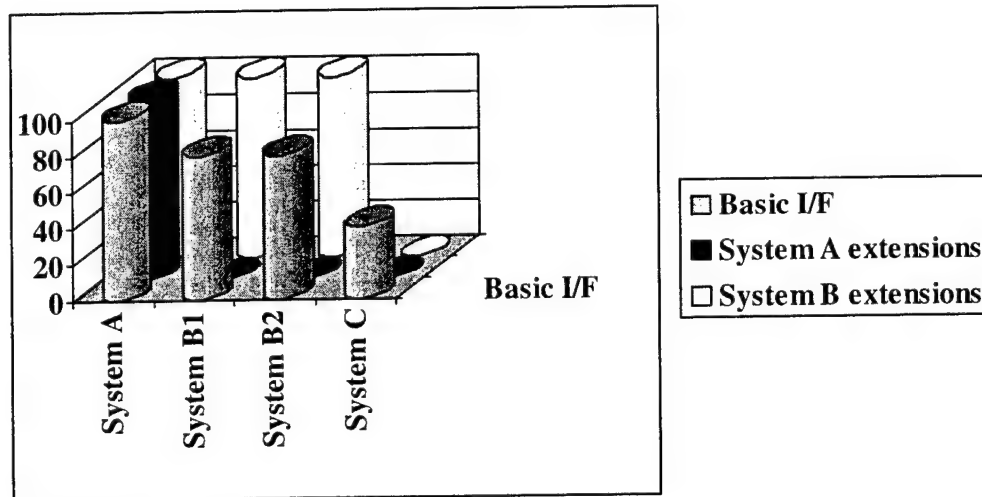


Figure 1: Coverage of Interfaces for Each System in Product Line

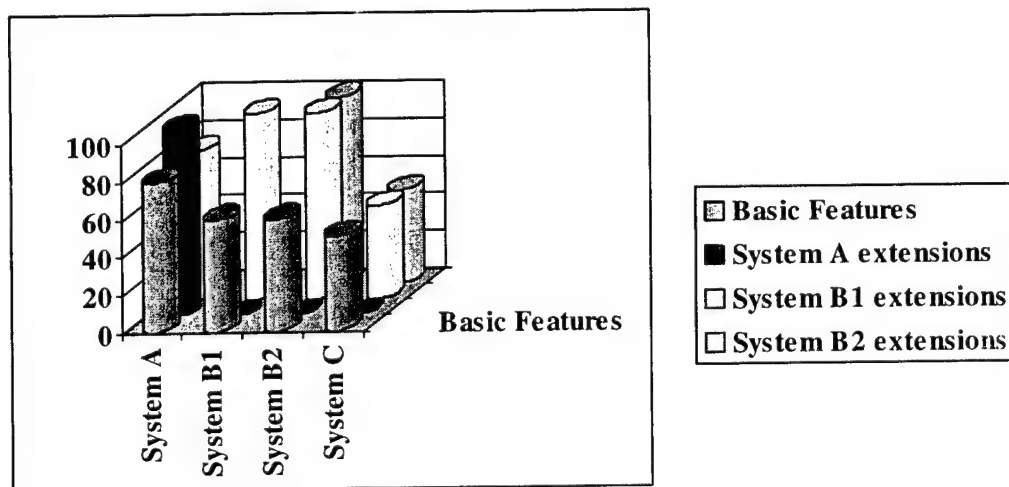


Figure 2: Coverage of Features for Each System in Product Line

The SPO established three goals for its product line approach:

1. Develop a core software asset base. It must address interfaces and features in terms of requirements, architecture, and components. It must also include a production plan to generate actual product line systems.
2. Develop an efficient system integration capability. The current approach requires too much rework in order to integrate new interfaces or features.
3. Develop a product line architecture that enables the SPO to add or change functionality with minimal impact to current user interfaces.

The SPO also developed lower-level objectives to measure the product line goals and to set time tables. For example, within one year, there must be a sufficiently rich asset base to construct a working sub-system prototype.

Based on these assumptions, the business case team then identified potential development scenarios (assets and products) and alternative product line strategies [Bergey 01]. These efforts helped assure higher management that all options were being considered and that a single strategic reuse decision would not be forced on them. These efforts also gave management insight into any proposed changes in terms of customer impact, costs, benefits, and time for implementation. The business case team ranked the alternative product line strategies by potential risks, benefits, and investment required.

Table 2 lists the alternative product line strategies.

Table 2: Candidate Alternative Courses of Action

Candidate Courses of Action (COA)		
ID	Descriptive NAME	Contractual Approach
COA #1	PRE-QUALIFY multiple contractors awarded contracts and collaborate in effort	Competitive award of 2 or 3 task order contracts to multiple offerors to collaborate in product line and product development (i.e., an avionics software system) in accordance with pre-established source selection criteria.
COA #2A	SOLE SOURCE to ACON: System A software contractor	Sole source award of a single task order contract to one of the incumbent software contractors to develop a software product line and a baseline product. These two mutually exclusive choices are explored to consider the impact of one legacy software contractor being selected over another.
COA #2B	SOLE SOURCE to BCON: System B software contractor	
COA #3	FLY-OFF resulting in award of single contract to best-value performer	Competitive award of two or three short term contracts. Have selected offerors compete in a "fly-off" development effort leading to the award of a single task-order contract to the offeror demonstrating "best value" performance.
COA #4	OPEN Acquisition single contract award	Competitive award of a single task order development contract to the offeror <sup>1</sup> submitting the "best value" proposal.

## 2.2 Estimate the Likely Costs and Analyze Potential Risks for All Alternatives

The organization analyzed the cost categories for funding four systems over the next five years. The costs were expressed as either start-up or steady state. Start-up costs included planning and analysis, as well as infrastructure and core asset development. Steady-state costs included product development, sustainment, and evolution costs.

Figure 3 shows the complete breakdown of costs from product line initiation to implementation.

<sup>1</sup> It is an offeror's prerogative to represent a single contractor or a contractor team.

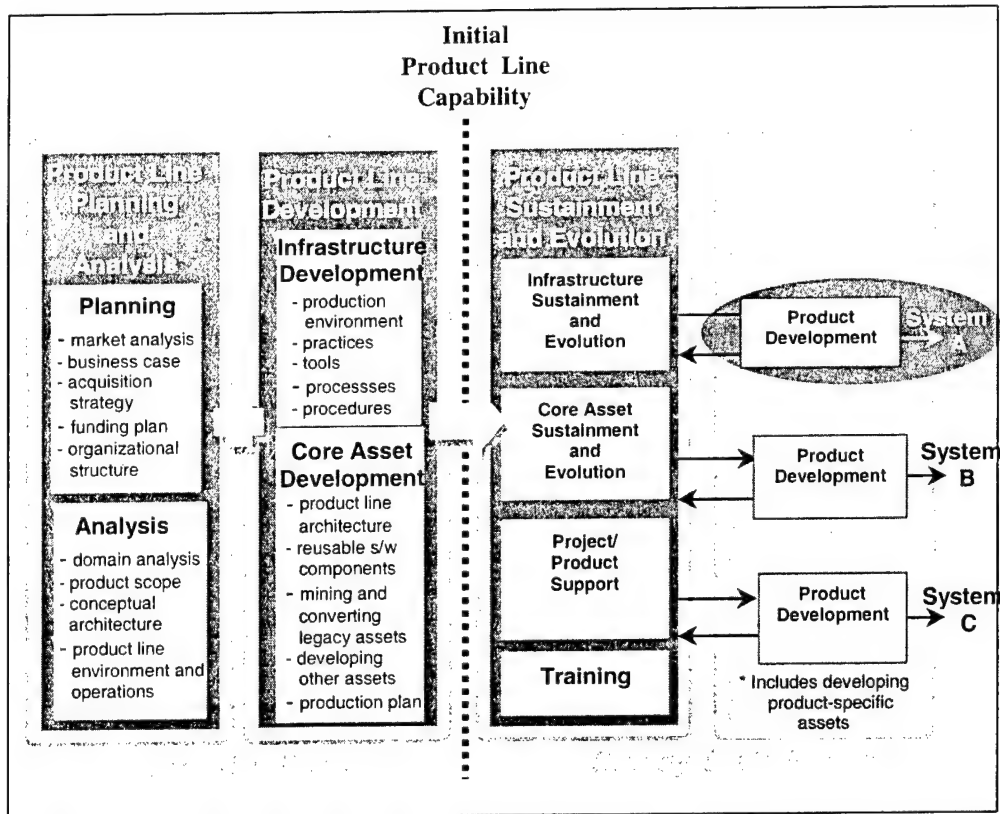


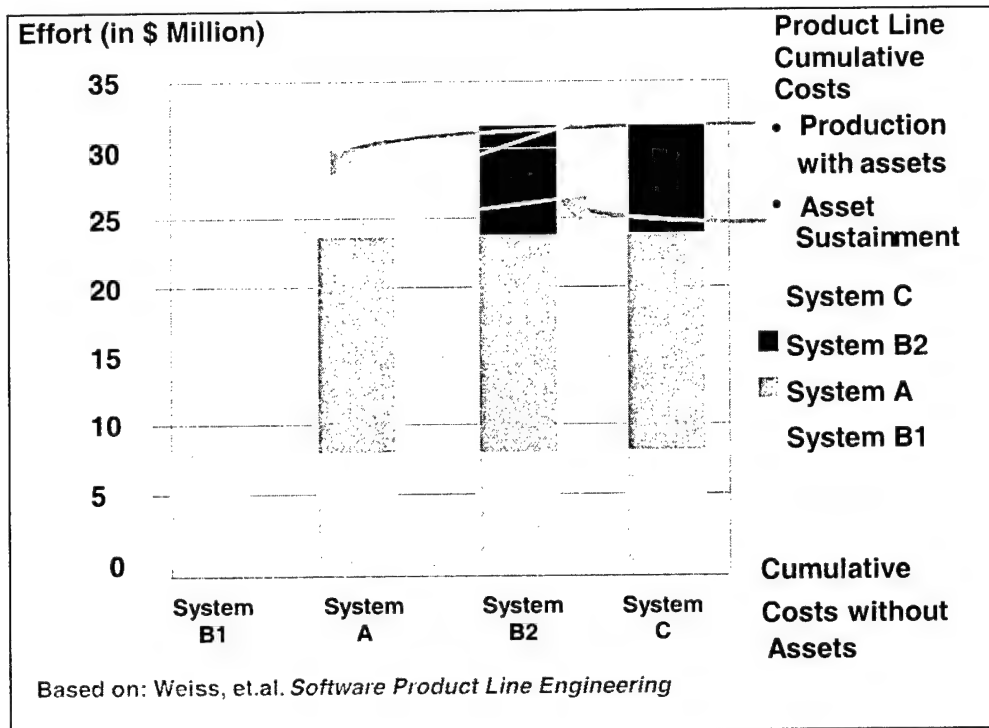
Figure 3: Breakdown of Costs to Field the DoD Product Line

Based on the initial scope of the product line, the business case team estimated and compared likely costs of each alternative product line strategy. Figure 4 shows cumulative cost estimates for four successive projects, with and without product line scenarios. Weiss describes the assumptions, methods, and rationale behind the estimation process [Weiss 99].

Without core assets, the expense of producing each successive project accumulates based on the cost of development from scratch, or with minimal reuse. The following conclusions can be drawn for the DoD product line shown in Figure 4.

- There is an initial cost to develop assets and move to a product line approach. It included the cost of training, incentives, and tool development or procurement, as well as the actual cost to develop assets. Start-up costs were estimated at \$23 million based on
  - historical data on hours/source lines of code
  - using lines of code delivered in legacy systems to predict size of asset base
  - development of first target system included in start-up
  - 10% investment in process improvement added to other development costs

- With each successive project, cumulative asset costs increase due to improvements in the assets or due to the effort in developing new assets. These are considered sustainment costs. The total sustainment cost was developed by adding 5% to the baseline cost for each successive application of the assets.



*Figure 4: Product Line Versus Conventional Development Costs*

The business case team estimated cumulative costs of a product line approach at \$36 million. This figure included startup and asset development and sustainment costs for all four projects. It did not consider the net present value of the up-front investment.

- asset baseline costs: \$23M (includes costs for training)
- asset sustainment costs for developing 4 systems: \$4M (5% of baseline cost per system)
- product development costs: \$3M per system after initial system (for a total of \$9M)

By comparison, the cost of developing the four systems independently was estimated at \$38 million. This figure was based on the total lines of code produced or modified times cost per line for new versions of the legacy systems.

These figures showed that the business case for a product line approach could not be made on cost savings alone. The relatively low costs for new versions of Systems B1, B2, and C

cancel out the much higher cost of System A. A cost-driven business case would require another new version of System A to prove the value of the product line approach.

As an alternative, the business case team considered phasing in product line assets, rather than incurring the total asset development cost up-front. This would lower start-up cost, but increase asset sustainment and further asset production costs. The phase-in approach was considered too risky. There was insufficient assurance that the asset base would grow to meet the demand.

Development and sustainment costs represent obvious ways to measure the worth of the product line approach. However, there are others. Once new versions of the legacy systems are fielded, each must be separately maintained. Over the next five years, systems B2 and C will add interfaces not present in earlier releases of Systems A and B1. Maintenance of A and B1 must include these interfaces.

Under the current development method, the SPO needed to contract with multiple organizations to deploy interfaces across each platform. Under the product line approach, common interfaces would be developed from the start. Therefore, the business case team compared long-term maintenance costs to make its case and achieve the SPO's goal of an efficient system integration capability. The team found the following:

- Maintenance for the four systems developed under the current method will cost \$38 million.
- Under the product line approach, the cost of maintenance will build by \$13 million per system over the same period.
- By the end of the four-year maintenance cycle, the product line approach will save the SPO \$25 million.

In short, the maintenance argument would justify the initial product line investment, if all systems are maintained for four years after initial deployment.

### **2.3 Estimate the Likely Benefits and Contrast With Current Business Practice**

Having identified cost components of the product line approach, the business case team next considered courses of action (COAs) and strategies. It analyzed organizational drivers such as the ability to leverage legacy software, responsibility for end product, schedule constraints, etc. Then it compared each driver to the product line strategies, listed in Table 2, and rated according to risk. The labels of Table 3, "R," "Y," and "G" represent varying degrees of risk from High ("R") to Low ("G") given each strategy and driver.



Table 3: Summary of Impact of Acquisition/Business Drivers on Candidate COAs

	PRE-QUALIFY	SOLE SOURCE		FLY-OFF	OPEN Acquisition
Business/Acquisition Drivers	COA #1	COA #2-A	COA #2-B	COA #3	COA #4
1. Contractor(s) will have access to all legacy software.	(Y)		(G)		
2. Contractor(s) will have relevant expertise be able to leverage GFI (ACON) software.	(Y)	(G)	(Y)	(G)	(Y)
3. Acquisition approach will have potential to involve both legacy system contractors.	(G)			(Y)	(Y)
4. Contractor(s) will be totally responsible for avionics software performance.		(G)	(G)	(G)	(G)
5. Acquisition approach will be compatible with schedule constraints.	(Y)	(G)	(G)	(Y)	(Y)
6. SPO can acquire government-use data rights for all product line assets.	(G)	(G)	(Y)	(G)	(G)
7. Acquisition approach will reduce risk of acquiring a product line capability.	(Y)	(Y)	(G)	(G)	(Y)
8. Likelihood of being able to avoid a protest that could impact acquisition schedule	(G)			(G)	(G)
9. SPO's perception of the affordability of approach (based on historical cost data)	N/A	(Y)	(G)	N/A	N/A

While these benefits are clearly less tangible than absolute cost, they will affect the sponsor's ability to acquire future systems quickly, efficiently, and affordably.

## 2.4 Develop a Proposal for Proceeding

The business case demonstrated the value of a product line approach in which systems share a common architecture and are built from components. Figure 5 shows an overview of this approach, one that is common to many product line endeavors. Depending on circumstances, the SPO may use multiple contractors to acquire the architecture and components. In each case, the SPO would use a Product Line Concept of Operations<sup>1</sup> to capture the details of its strategy.

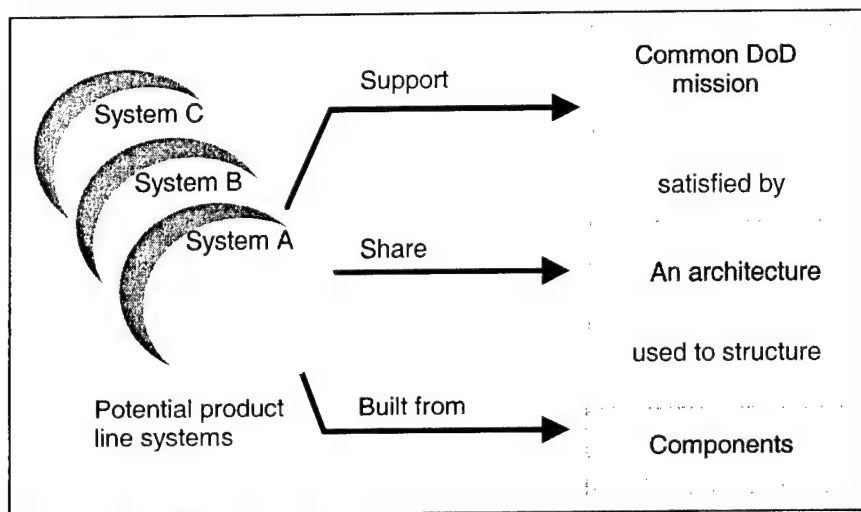


Figure 5: Vision for Product Line

## 2.5 Close the Deal: Make Final Adjustments and Proceed to Development

The business case study group recommended that the SPO invest in product line acquisition using integrated product teams (IPTs). Specifically, the group suggested that the SPO follow these six steps:

1. Under existing contracts, the SPO should task the legacy contractors to work jointly to define a conceptual architecture for the software product line and develop a requirements specification for a common DoD product line.
2. In parallel with Task 1, the SPO should create a Product Line/Architecture IPT. It should define the product scope, generate the conceptual architecture, develop the requirements specification, and gather the documents to be provided as government-funded information (GFI).
3. The Product Line/Architecture IPT should use the deliverables produced under Tasks 1 and 2 to develop a comprehensive specification. This will reduce the risk of having to

<sup>1</sup> A Concept of Operations (CONOPS) document defines the organization's product line approach. The CONOPS guides the organization as it plans and executes the process of fielding a product line, from product line scoping, through architecture, component development, and product development [Cohen 99].

amend the request for proposal (RFP) and enhance the likelihood of acquiring a higher quality product.

4. In parallel with the preceding tasks, an Acquisition IPT should develop and execute a competitive procurement plan. The group should refine the acquisition strategy, describe the acquisition tasks, and develop a detailed schedule with measurable milestones. (This will help the SPO to stay focused, to meet its RFP schedule, and to achieve the organization's overall acquisition goals and objectives.)
5. The Acquisition IPT should integrate the enhanced work products developed by the Product line/Architecture IPT into the RFP, complete RFP sections for requirements and selection criteria, and develop a source selection plan. (This will enable the contracting officer to begin the formal solicitation.)
6. A Source Selection Team should evaluate the offerors' proposals according to the plan developed in Task 5. The Team should award a single contract to develop a product line capability and common architecture. The contract will also require the contractor to follow the production plan and deliver a new version of System A. (This will complete the award phase. At this point, the contract administration and performance phase will begin.) Having both a rigorous acquisition plan and a judicious source selection plan will reduce the risk of an offeror filing a bona fide protest.

---

## **3 Business Case Results**

### **3.1 Conclusions**

The business case included a set of recommendations and cost/benefit analyses. They helped the SPO to determine the merits of a product line approach, based on the following considerations:

#### **GOAL 1: Reduction in costs**

The case for the product line cannot be made on direct development costs alone. Over the next five years, total costs for developing the product line assets and the initial set of product line systems will not be substantially lower than that for developing stand-alone systems. However when combined with product line sustainment costs over eight or more years, the product line approach will provide substantial savings, based on its ability to share interfaces among various products.

#### **GOAL 2: Increased reliability**

The business case team developed acquisition options to promote competition and to increase the SPO's ability to measure and to compare product quality. This approach should yield a more reliable set of systems than the current sole contractor approach. In addition, sustaining a common set of assets has shown to reduce defects and improve overall system performance.

#### **GOAL 3: Ease in the integration of new interfaces**

The common product line architecture was specifically addressed to meet this goal.

### **3.2 Lessons Learned**

1. Establish an Integrated Product Team (IPT) to develop the business case.
2. Include technical, marketing, and acquisition representatives on the IPT.
3. Allow a time frame of four to six weeks to gather the information, develop and compare alternative strategies, and actually write the business case. (This period can be greatly shortened if the organization routinely maintains cost and effort data from completed projects and other business cases.)
4. Capture and make available accurate historical data to support the business case process.

---

## References

- [Bergey 01] Bergey, J. K.; & Goethert, W. B.; *Developing a Product Line Acquisition Strategy for a DoD Organization: A Case Study* (CMU/SEI-2001-TN-021). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. [online] Available WWW: <URL: <http://www.sei.cmu.edu/publications/documents/01.reports/01tn021.html>> (April 2001).
- [Clements 00] Clements, P.; & Northrop L. M. *A Framework for Software Product Line Practice*. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. WWW: <URL: <http://www.sei.cmu.edu/plp/framework.html>> (2000).
- [Cohen 99] Cohen, Sholom. *Guidelines for Developing a Product Line Concept of Operations* (CMU/SEI-99-TR-008 ADA367714). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. [online] Available WWW: <URL: <http://www.sei.cmu.edu/publications/documents/99.reports/99tr008/99tr008abstract.html>> (1999).
- [Humphrey 00] Humphrey, Watts. "Justifying a Process Improvement Proposal." news@sei interactive, Available WWW: <URL: [http://interactive.sei.cmu.edu/news@sei/columns/watts\\_new/2000/march/watts-mar00.htm](http://interactive.sei.cmu.edu/news@sei/columns/watts_new/2000/march/watts-mar00.htm)> (March 2000).
- [Weiss 99] Weiss, David M. et al. *Software Product Line Engineering*. Reading, MA: Addison-Wesley, 1999. Pg. 45-49.



REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503				
1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE April 2001		3. REPORT TYPE AND DATES COVERED Final	
4. TITLE AND SUBTITLE Case Study: Building and Communicating a Business Case for a DoD Product Line			5. FUNDING NUMBERS F19628-00-C-0003	
6. AUTHOR(S) Sholom Cohen				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213			8. PERFORMING ORGANIZATION REPORT NUMBER CMU/SEI-2001-TN-020	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES				
12A DISTRIBUTION/AVAILABILITY STATEMENT Unclassified/Unlimited, DTIC, NTIS			12B DISTRIBUTION CODE	
13. ABSTRACT (MAXIMUM 200 WORDS) This case study describes a US Department of Defense (DoD) weapon system development effort and compares the current way of developing software systems to the product line approach. The report describes a product line business case acquisition context, alternative development paths, and a comparison of their merits. While the report "sanitizes" the actual organization and product line, it typifies the process of building a product line business case.				
14. SUBJECT TERMS business case study, product line practice, software product line, software system acquisition			15. NUMBER OF PAGES 29	
16. PRICE CODE				
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	